APPENDIX A

Data Kernel Specification List:

MODEM FUNCTIONS:

Data Switch Selector

DESCRIPTION: This selector is capable of selecting antenna payload data based on antenna number and routing the payload to the appropriate receiver. The inputs to the data switch selector are the data from the antenna bus, which is tagged with carrier information and sector information. With C carriers and S sectors the total number of input streams is C*S. These can be mapped among any of the C*S outputs ports.

INPUTS: C*S data streams

OUTPUTS: C*S data streams

PARAMETERS: configuration number $\in \{1,2,\ldots C*S\}$

Sample Interpolation

DESCRIPTION: This function takes as input IQ samples at 2fc, 4fc, or 8fc, and based on the value of INTERP_MODE either performs interpolation or passes the data through. If interpolation is performed, INTERP_MODE specifies the desired interpolation factor to achieve IQ samples at 8fc at the output of the block.

INPUTS: Input samples.

OUTPUTS: Output samples.

PARAMETERS: INTERP_MODE interpolation factor,

INTERP_FIL interpolation filter (number of taps, tap coefficients)

Sample-Epoch Selection

DESCRIPTION: This function selects one of 8 sampling phase epochs, denoted by CHIP_SELECT_EPOCH of an effective fc sampling clock on the 8fc input IQ stream to produce an IQ sample stream at fc. OFFSET_MODE is used to specify if subchips must be offset to receive OQPSK.

INPUTS: Input samples.

OUTPUTS: Output samples.

PARAMETERS: OFFSET MODE offset mode

CHIP_SELECT_EPOCH sample epoch

Matched Filter

DESCRIPTION: This block will implement a code (or sub-code) matched filter. It may be used in either a Multi-standard CDMA traffic channel receiver or a RACH-type receiver (3GPP). Both coherent and noncoherent accumulation modes are allowed.

INPUTS:

RxData_I Real part of the on time output of the chip-matched filter

RxData_Q Imaginary part of the on time output of the chip-matched filter

OUTPUTS:

Main_I Real part of matched filter output

Main_Q Imaginary part of matched filter output

PARAMETERS:

Type of accumulation	coherent or noncoherent

Multi-standard Despreader

DESCRIPTION: This block will implement a Multi-standard despreader/descrambler. It will support all the code/data modulation formats used in IS-95, IS2000, ARIB, and 3GPP. These include BPSK/QPSK modulation for data or spreading code, and single or dual channel QPSK for spreading code. It can also be configured for a user-specified coding scheme. Both real and complex despreading /descrambling are supported. The code input is obtained from a Multi-standard code generation unit that may be configured to output either short or long codes. The data inputs to this block are obtained as the early, on time and late sampling epochs. Both coherent and non-coherent accumulation modes will be allowed.

INPUTS:

RxData_I	Real part of the on time output
RxData_Q	Imaginary part of the on time
Early_I	Real part of the early output
Early_Q	Imaginary part of the early output
Late_I	Real part of the late output
Late_Q	Imaginary part of the late output
Short_I	Real part of short code for IS-95, IS2000, ARIB, or 3GPP
Short_Q	Imaginary part of short code for IS-95, IS2000, ARIB, or 3GPP

Long Code — Input bus for de-channelization with IS-95B long code or IS-2000 channelization codes, or 3GPP channelization codes.

OUTPUTS:

Main_I — An output bus containing the real part of the despread data

Main_Q — An output bus containing the imaginary part of the despread data

Early_I — This output will be used in the delay locked loop circuit for time tracking of the spreading sequences.

Early_Q — Description same as above

Late_I — Description same as above except that it is the sample selected at an epoch after the on-time sample

Late_Q — Description same as above

PARAMETERS:

Mode — IS-95, IS2000, ARIB, 3GPP-FDD, 3GPP-TDD, 3GPP2, or user-defined.

Configuration — Correlate or Pipeline Correlate

Type of accumulation — If matched filter, coherent or noncoherent

No. of segments — If noncoherent, number of segments within the correlation interval

Multi-standard Dechannelizer

DESCRIPTION: This function takes as input IQ samples at fc and performs dechannelization (MODE_ON) according to a specified mode (MODE_CODE_CHANNEL) using a reference input

spreading code (REFER_CHAN_SOURCE_I, REFER_CHAN_SOURCE_Q). Typically, this is used for despreading using Long code or Orthogonal Variable Spreading Factor (OVSF) codes.

INPTUS: Complex despread data for on-time, early, late

OIUTPUTS: Complex despread and dechannelized data for on time, early, late

PARAMETERS: mode

Code Generation Unit

DESCRIPTION: This block provides all required codes among a set of standards including {IS-95, CDMA2000, IS2000, UTRA, ARIB, 3GPP}. Various codes are generated for both the uplink and downlink requirements. This block also contains timing information for the modem and for each individual finger of a RAKE receiver. The code generation unit also contains a Mask Generation Unit, which is used to transform a given offset into a set of code dependent parameters. The parameters are used in the reassignment of a code's phase.

INPUTS: None

OUTPUTS: Pseudo-random Noise code sequences for the downlink and each RAKE finger.

PARAMETERS: mode of operation, search slewing, search reset, reference timing adjustment, sequence reassignment parameters including {sequence number, OVSF number, offset}

Integrate-and-Dump

DESCRIPTION: This function takes IQ samples at fc and performs (MODE_IDUMP) an accumulation over INT_LEN chips, producing outputs at fc/INT_LEN.

INPUTS: Input samples.

OUTPUTS: accumulated result samples.

PARAMETERS: integration length from the set {4,8,16,32,64,128,256}

Multi-standard Searcher Control

DESCRIPTION: This function processes correlation results, performs peak detection, threshold comparison, and controls a Multi-standard multi-dwell search engine.

INPUTS: one frame of received data

OUTPUTS: a detection signal.

PARAMETERS:

Maximum number of peaks to detect,

search engine microcode (number of dwells, integration length for each dwell, threshold for each dwell, search window size, state transition control).

Transport Format Decoder

DESCRIPTION This block is used in systems where the network operator elects to use transport format indicators to denote the transport format of the accompanying dedicated traffic channels. In 3GPP, for example, the TFCI bits in each uplink frame represent the number of different transport formats multiplexed within the

frame under consideration. The TFCI bits are encoded in the mobile using a 10-30 punctured Reed-Muller code and interleaved across the entire frame. This block will decode the received TFCI codeword in the DPCCH channel of each frame and output the 10-bit TFCI word that may be used for transport format combination and DPDCH spreading factor computation.

INPUTS:

Code	A 30 bit codeword representing a punctured Reed-Muller codeword

OUTPUTS:

DEC_TFCI	A 10 bit word representing the transport format combination

PARAMETERS: None

Dynamic Spreading Factor Computer

DESCRIPTION: Rate matching of multiplexed composite coded traffic channels in some of the newer WCDMA standards, such as 3GPP uplink, is performed using a combination of semi-static and dynamic rate matching mechanisms. On a frame-by-frame basis, depending on the total number of bits in the CCTrCH, spreading factor is selected from an allowed set of values i.e. single physical channel with SF = 256 down to 6 multicode channels with SF = 4. This block implements spreading factor determination from input TFCI bits indicating the transport format for the current frame. If possible, the algorithm selects only one physical channel for

transmitting all the multiplexed bits within the CCTrCH. If the total number of bits is in excess of what is required to avoid multicode transmission, then puncturing is used.

INPUTS:

TFCI	10-bit TFCI used to determine the transport format and DPDCH spreading factor

OUTPUTS:

SF	DPDCH spreading factor for the current frame

PARAMETERS: None

Fast Hadamard Transform

DESCRIPTION: This block performs a Fast Hadamard Transform based on the input vector of length N_FHT

INPUTS: input samples.

OUTPUTS: output samples.

PARAMETERS: Basis Vectors (coeffs)

Energy Estimator

DESCRIPTION: This block applies only to coherent mode of operation where known pilot bits are transmitted. A finger measures the power delay profile for a particular path during a time slot when pilot bits are available and the result is the instantaneous channel measurement. This result is low pass filtered with a first order filter and provides the energy estimate. This block is enabled or disabled according to the mode of operation of the receiver.

INPUTS: Pilot symbols within a particular time slot

OUTPUTS: Energy estimate for the path

PARAMETERS: The pole for the first order filter

Timing Parameter Estimator

DESCRIPTION: The purpose of the timing parameter estimation block is to maintain the timing (PN code) synchronization with the Transmitter (Remote device). The function performs the instantaneous time estimation of the received signal (PN code) stream, which gets processed via a loop filter to produce an error signal to drive a VCO (or NCO). The output of VCO (or NCO) is used to correct the locally generated PN code time or the on-time sample selection process. In case of the integer (or more) chip time delay the CGU unit is either advanced or retarded. Similarly the epoch of the on-time arrival sample are advanced or retarded to compensate for any residual fractional time delay. This timing correction allows for the synchronization of the chip timing with the remote device. The error signal can be produced coherently or non-coherently depending on the system requirements and characteristics and is programmable from the external controller device. The loop filter parameters can also be controlled to affect the loop bandwidth, locking range etc.

INPUTS:

E_I, E_Q, L_I, L_Q Early/Late Complex signal from the Sample Select block

THETAHAT_PILOT Pilot (Phase) estimate

OUTPUTS:

TI_INTG, TI_FRAC Control Signal to compensate for fractional and integer chip (time) delay

EPOCH_SELECT Sample Select Location

CGU_THROTTLE CGU integer advance/retard

PARAMETERS:

E_TYPE Type of error computation (Coherent or Non-coherent)

NSAMP_NONC Number of samples for Non-coherent error estimation

Loop Filter Parameters

LOPFIL_ORD Order of the filter

LOPFIL_COEF Coefficients of the Loop Filter (based on the order of filter)

LOPFIL_UPDR Loop Up-date Rate

Channel Estimator

DESCRIPTION: This block applies only to coherent mode of operation where known pilot bits are transmitted. A finger measures the power delay profile for a particular path during a time slot when pilot bits are available and the result is the instantaneous channel measurement. This slot's result is filtered with a K-tap FIR filter where the taps correspond to measurements in previous slots. Each slot measurement is weighted and combined according to a set of tap weights. The final result is the weighted multi-slot average (WMSA) and is used for coherent de-rotation in the receive

path. The number of weights (IE slots) ranges from 1 to K. This block is enabled or disabled according to the mode of operation of the receiver.

INPUTS: Pilot symbols within a particular time slot

OUTPUTS: WMSA complex result

PARAMETERS: An array containing the set of weights.

Rotator

DESCRIPTION: Rotates in the complex plane the input1 by the angle associated with input2 with scaling.

INPUTS: input samples (I&Q), scaled phase estimate.

OUTPUTS: output samples.

PARAMETERS: None.

Alignment/Deskewing

DESCRIPTION: This function aligns in time the received multipath components from a set of N fingers.

INPUTS: input samples, number of paths to align, misalignment on each path.

OUTPUTS: output samples.

PARAMETERS: None.

Combiner

DESCRIPTION: This block performs configurable weighting and combining output streams. Each finger is assumed to track a single multi-path and a set of fingers are set up for combining. After time alignment of each stream, the set is combined. In non-coherent IS-

95 mode of the receiver, for example, the Hadamard Transform results are arithmetically summed together before the soft decision device. In coherent-mode, each finger provides an estimate of instantaneous channel energy and each is selected for combining base on a sufficiently large SIR. The largest and smallest instantaneous estimate is recorded for each finger over a window of size K. The term " S over-bar" is the energy estimate. The terms Δ_{noise} and Δ_{rake} are design parameters. The finger is chosen for combining if the following condition is true:

$$\bar{S}_l(k) \geq \max\left\{S_{\min}(k)\cdot 10^{0.1\Delta_{noise}}, S_{\max}(k)\cdot 10^{-0.1\Delta_{rake}}\right\}$$

INPUTS: All logical channels for all multi-paths

OUTPUTS: Multi-path combined logical channels

PARAMTERS: mode of operation coherent or non-coherent, Δ_{noise} level, and Δ_{rake} level

Soft Decision Computer

DESCRIPTION: This block applies only to non-coherent mode where channelization is similar to that of IS-95-type CDMA systems. The soft decision device transforms a 64-ary vector into a 6-ary vector. The transformation is performed after a Hadamard Transform and used in decoding the orthogonal data on the IS-95 "reverse link". This block is enabled or disabled according to the mode of operation of the receiver.

INPUTS: A 64-ary vector

A 6-ary vector representing the soft decision.

OUTPUTS:

PARAMETERS: none

Interpath Interference Equalizer

DESCRIPTION: This block implements a linear minimum mean-squared error equalizer for suppression of interpath interference in high data rate, low spreading factor transmission, where interpath interference dominates over multi-access interference. Only post-combining linear MMSE equalizer will be considered i.e. equalization is performed after despreading. Incorporation of nonlinear equalizers using data decisions to cancel ISI in future symbols is TBD.

INPUTS:

In Complex output of the despreader followed by multipath combiner

OUTPUTS:

Out IPI suppressed signal to be used as soft decision for channel decoder

PARAMETERS:

Number of taps This parameter determines the number of symbols over which ISI is equalized and should correspond to the maximum delay spread of the multipath channel

Mu LMS adaptation step size

Receive Antenna Diversity Combiner

DESCRIPTION: Performs user-specified diversity selection of 4 largest paths from 8 antenna paths.

INPUTS: input samples from different sources.

OUTPUTS: output samples.

PARAMETERS: selection method (sum(max(a1,a2)), sum(max(a1)+max(a2)), or pass through(a1+a2), Note, a1, a2 are paths from 2 antennas).

<u>CODEC FUNCTIONS:</u>

DeInterleaver Controller

DESCRIPTION: This takes the deinterleaved data from the external RAM and transfers it to the input buffers of the appropriate channel decoding kernel, using either DMA or a user routine to perform scaling on the data input. This routine may also perform depuncturing (using don't care bit insertion) for puncturing patterns not currently supported by the hardware.

INPUTS: Deinterleaved data in external RAM

OUTPUTS: Deinterleaved, channel ordered data located in decoding kernel input buffers

PARAMETERS:

Input_Data_Scaling(),

Channel_Types[],

Buffer_RAM_Allocation[],

Decoding_Kernel_Assignment[]

Turbo Decoder

DESCRIPTION: This block performs depuncturing and iterative Turbo decoding on the encoded input data to produce an estimate of the input data. Note that all the intermediate results (i.e. Extrinsic Out/In between the component MAP decoders) will be visible in the Buffer_RAM. The block contains a lookup table for Log-MAP correction, which may be reprogrammed by the user. Also note that the use of on-the-fly hardware depuncturing enables the decoder to use a RAM block size equivalent to the maximum input block size (prior to depuncturing), which will be significantly less than the decoding block size.

INPUTS: Encoded data to be Turbo decoded (RATE blocks, each of length BLOCK_LENGTH), within the Buffer_RAM

OUTPUTS: Decoded Data (length BLOCK_LENGTH),

PARAMTERS: BLOCK_LENGTH, CONSTRAINT_LENGTH, RATE, POLYNOMIALS[], MAX_NO_ITERATIONS, MAP_DECODER_SELECT, PUNCTURING_SELECT, Iteration_Terminate_Routine(), Input_Scaling_Routine(), Output_Handling_Routine(), Map_Correction_Table[]

Convolution Decoder

DESCRIPTION: This block performs depuncturing and Viterbi convolution decoding on the encoded input data to produce an

estimate of the input data. Note that all the intermediate results (i.e. previous/ next state memory and traceback memory) will be visible in the Buffer_RAM. Also note that the use of on-the-fly hardware depuncturing enables the decoder to use a RAM block size equivalent to the maximum input block size (prior to depuncturing), which will be significantly less than the decoding block size.

INPUTS: Encoded data to be convolutionally decoded (RATE blocks, each of length BLOCK_LENGTH), within the Buffer_RAM

OUTPUTS: Decoded Data (length BLOCK_LENGTH),

PARAMETERS: BLOCK_LENGTH, CONSTRAINT_LENGTH, RATE, POLYNOMIALS[], PUNCTURING_SELECT, OUTPUT_SD_BIT_WIDTH, Input_Scaling_Routine(), Output_Handling_Routine()

<u>SEQEUNCING OF MODEM AND CODEC FUNCTIONS:</u>

RECEIVE PATH

A Receive path can be realized by configuring the kernels according to the type of operation desired, inputs, outputs, parameters, and initiation and termination conditions. From the blocks described above, the following typical receive path can be realized.

Per CDMA Modem Signal Processor Chip:

1. Data Switch Selector: routing of antenna payload data

Per Antenna:

2. Sample Interpolation: configurable interpolation of input data (which may arrive at 2fc,4fc) to an output rate of 8fc

Per Antenna Per Finger

3. Sample Epoch Selection: selection one sample epoch of 8fc data per chip period

4. Generic Despreading/Descrambling with Short/Long Codes: configurable code-generation and descrambling of input data based on code/data modulation format (coherent/noncoherent; single/dual channel, BPSK/QPSK data/code demodulation)
    - Add despreader Requirements: a) MF mode (shared between fingers) b) partial correlation accumaltion c) non/coherent accumumalation

5. Generic Dechannelization with Walsh/OVSF Codes: configurable code-generation and dechannelization of input based on Walsh/OVSF code modulation

6. Searcher Control with peak/threshold detection
7. Transport Format Indicator Decoder (Reed Muller)
8. Dynamic Spreading Factor Computer (used, for example, in 3GPP from decoded TFCI bits)
9. Code Generation Unit
    - Add: everything in current spec
    - Add: extended s(2) codes for 3GPP

10. Integrate-and-Dump : integration over parameterizable number of samples

11. Fast Hadamard Transform: perform Fast Hadamard Transform using computationally efficient technique (soft decision out), includes chip-symbol storage.

12. Energy Parameter Estimation: performs configurable accumulation of symbol energies over window for RSSI and AGC computations

13. Timing Parameter Estimation: performs configurable delay-locked loop tracking of signals for both non-coherent and coherent reception techniques

14. Channel Estimation: performs configurable estimation of channel parameters, using user selectable control to realize one-shot CIR estimation and weighted multislot averaging techniques
15. Magnitude: performs magnitude operation on real and imaginary parts
16. Soft-Decision Device – used to achieve optimum reception in combining FHT outputs in IS95 and derivative system by using user-programmed 64-ary to 6-ary selection.
17. Automatic Gain Control (for power control, measure SIR and Eb/N0)
18. Rotation: performs configurable or auto symbol rotation and scaling
19. Alignment /Deskewing- performs configurable alignment of finger outputs
20. Combiner- performs configurable weighting and combining of finger output

21. Equalization- performs configurable LMMSE equalization for channels with low-spreading factor and high inter-path interference per finger output

Per Antenna-Pair
22. Receive Antenna Diversity: performs selection of N largest paths from M antenna-paths via
- Add: this is post processing antenna diversity
- The receiver BTS examines 8 received paths 4 (Ant 1), 4 (Ant2), select 4 largest

23. Preprocessing Receive Antenna Diversity: this is pre-processing diversity with LMS derived weighting and MMSE adaptation of the weights per path
- AntennaArray: The kernels can perform an MMSE-based combining of antenna returns, performed per path, per user channel. This requires MMSE weighting, RAKE combining, and remodulation of the RAKE output pilot decision into the MMSE for adaptation.

TRANSMIT PATH

Per User Channel:
1. Generic Scrambling with Short/Long Codes: configurable code-generation and data/code scrambling
2. Puncturing Mode Control: puncturing and insertion of data

Per User Channel Per Antenna:

3. Transmit Diversity Modes: performs user-specified open-loop and closed-loop transmit diversity schemes via Open-Loop and Closed-Loop transmit diversity methods:

   Open-Loop Methods:
   - TSTD (time-switched transmit diversity): antenna hopping with transmission alternating between antennas
   - OTD (orthogonal transmit diversity): orthogonal transmit diversity- transmission of two symbols after S/P conversion to parallel antenna paths
   - STTD (space-time transmit diversity): transmit transformed symbols (orthogonal)

   Closed-Loop Methods (employ selection transmit diversity-STD):
   - STD Mode 1- BS sends PCCPCH pilot bits, mobile measures and indicated beam selection; effective for low-mobility environments
   - STD Mode 2- BS cophases transmitted signal for coherent combining at mobile; channel estimated at mobile via dedicated channel
4. Generic Channelization with Walsh/OVSF Codes/Walsh Covering: configurable code-generation and channelization based on Walsh/OVSF code modulation
5. Generic IQ Spreading and Scrambling: configurable code-generation and IQ code modulation
6. FIR Filtering (Spectral shaping)
7. Rate-Dependent Scaling

Per Antenna:

1. Antenna-Sector Bus Muxing and Combining: selection, combining, and adding all channels per carrier-sector